

FAST BINARY EMBEDDING VIA CIRCULANT DOWNSAMPLED MATRIX – A DATA-INDEPENDENT APPROACH

Sung-Hsien Hsieh^{*,**}, Chun-Shien Lu^{*}, and Soo-Chang Pei^{**}

^{*}Institute of Information Science, Academia Sinica, Taipei, Taiwan

^{**}Graduate Inst. Comm. Eng., National Taiwan University, Taipei, Taiwan

ABSTRACT

Binary embedding of high-dimensional data aims to produce low-dimensional binary codes while preserving discriminative power. State-of-the-art methods often suffer from high computation and storage costs. We present a simple and fast embedding scheme by first downsampling N -dimensional data into M -dimensional data and then multiplying the data with an $M \times M$ circulant matrix. Our method requires $O(N + M \log M)$ computation and $O(N)$ storage costs. We prove if data have sparsity, our scheme can achieve similarity-preserving well. Experiments further demonstrate that though our method is cost-effective and fast, it still achieves comparable performance in image applications.

Index Terms— Circulant matrix, Dimensionality reduction, Embedding, Random projection

1. INTRODUCTION

1.1. Background and Related Work

Embedding of high-dimensional data into low-dimensional space is an important task in diverse fields due to the concern of computation and storage costs. In particular, embedding input data into binary space while preserving similarity is becoming popular because binary codes only require calculating Hamming distance implemented by adds.

Most existing techniques can be classified into two cases: data-independent and data-dependent. Data-independent techniques are popular due to their low-resource requirement and simplicity but often fail to give the best performance. On the contrary, data-dependent techniques often has better performance. But, along with the increase of size of data [1, 2], they are prohibited from being applied to learning because of high computation and storage costs.

In data-independent techniques, the popular and pioneered techniques are Locality Sensitive Hashing (LSH) [3] and its extension Shift-Invariant Locality Sensitive Hashing (SKLSH) [4] wherein embedding is based on random projection to achieve similarity-preserving. In [5], dimensionality reduction inherent in compressive sensing is exploited via random projection for image hash design. Gong *et al.* [6]

proposed a bilinear projection to further reduce computation and storage overheads during embedding. Chang *et al.* [7] proposed using a circulant matrix for projecting data because projection can be speeded up by Fast Fourier Transform (FFT). A learning mechanism is also considered in [6][7].

As for data-dependent techniques, different optimization criteria are used in the learning phase. For example, Iterative Quantization (ITQ) [8] aims to minimize quantization error after PCA. [9] proposed a sparsity regularizer in learning to reduce computation cost. Recently, deep neural network (DNN) [10] is used to jointly learn features and binary codes simultaneously. These methods learn compact codes especially for low-dimensional embedding. But, most of them require $O(N^2)$ computation and storage costs that may not be practical. Online learning is another issue along with increase of data [1][2].

1.2. Contributions of This Paper

In this paper, we propose a data-independent approach, including two steps: downsampling N -dimensional data into M -dimensional data first and then multiplying the data with an $M \times M$ circulant matrix. The proposed method, achieving $O(N + M \log M)$ in computation cost and $O(N)$ in storage cost, obviously outperforms state-of-the-art methods. Although our method and [7] are conceptually similar by introducing a circulant matrix for binary embedding, the major differences include: (i) We use downsampling matrix to compress the signal first, leading to the fact that the size of our circulant matrix can depend on M only instead of N . In [7], whatever M is, it requires the same computation cost $O(N \log N)$ because of using FFT for speeding computation. Thus, when $M \ll N$, [3][6] are even faster than [7]. (ii) We theoretically prove that even though downsampling is used, by combining downsampling with randomization, similarity-preserving is still satisfied well.

In addition to the fact that the computation and storage costs of our method are smaller than those of previous methods, experimental results reveal that their performances in image applications are comparable.

2. NOTATIONS

We display a matrix or a vector as bold. Let \mathbf{V} be a matrix, where \mathbf{v}_i is the i^{th} column of \mathbf{V} and \mathbf{v}^j is the j^{th} row of \mathbf{V} . $(\mathbf{V})_{i,j}$ is the $(i,j)^{th}$ entry of \mathbf{V} . Let $\mathbf{u} \in \mathbb{R}^N$ be a vector and let $\text{circ}(\mathbf{u})$ be a circulant matrix generated based on the seed vector \mathbf{u} . For example, for $\mathbf{U} = \text{circ}(\mathbf{u})$, the first row is $[(\mathbf{u})_0, (\mathbf{u})_1, \dots, (\mathbf{u})_{N-1}]$, the second row is $[(\mathbf{u})_{N-1}, (\mathbf{u})_0, \dots, (\mathbf{u})_{N-2}]$, and the last row is $[(\mathbf{u})_1, (\mathbf{u})_2, \dots, (\mathbf{u})_0]$.

3. PROPOSED METHOD

We first describe how to design a data-independent projection matrix to achieve both the lowest computation and storage costs in the literature. Then, we prove that the proposed method still satisfies similarity-preserving property. In this paper, following [3][7], similarity is measured as the angle between two vectors in the input data space.

3.1. Construction of Projection Matrix

The core idea is to design a projection matrix composed of a downsampling matrix and a circulant matrix achieving: (i) $O(N + M \log M)$ operations for fast embedding process. (ii) $O(N)$ bits for saving the projection matrix. (iii) Angle-preserving after embedding.

Binary embedding or 1-bit compressive sensing [11] is defined as:

$$\mathbf{h} = \text{sign}(\mathbf{A}\mathbf{x}). \quad (1)$$

where $\mathbf{x} \in \mathbb{R}^N$ is an input signal, $\mathbf{h} \in \mathbb{R}^M$ is the corresponding binary code, $\text{sign}(\cdot)$ is a sign function, and $\mathbf{A} \in \mathbb{R}^{M \times N}$ is a projection matrix defined as:

$$\mathbf{A} = \mathbf{D}\Phi\mathbf{R}. \quad (2)$$

Specifically, \mathbf{R} is either a uniform random permutation matrix (global randomizer) or a diagonal random matrix (local randomizer) whose diagonal entries $(\mathbf{R})_{i,i}$ are i.i.d Bernoulli random variables with equal probability. In our paper, \mathbf{R} implements both global randomizer and local randomizer simultaneously¹. $\Phi \in \mathbb{R}^{M \times N}$ is a downsampling matrix with $(\Phi)_{i,j} = 1$ if $(-i+j) \bmod M = 0$ for $0 \leq i, j \leq N-1$. $\mathbf{D} = \text{circ}(\mathbf{d}^0) \in \mathbb{R}^{M \times M}$ is a circulant matrix with seed vector \mathbf{d}^0 , where \mathbf{d}^j is the j^{th} row of \mathbf{D} , to achieve: 1) faster computation than traditional random matrix; 2) fairly spreading the information into each bit.

Based on Eq. (2), the computation cost includes (i) \mathbf{D} is implemented by FFT with $O(M \log M)$. (ii) Φ , in fact, acts to downsample $\mathbf{R}\mathbf{x}$ and cost $O(N)$ adds and zero multiplications. (iii) Each column in \mathbf{R} only has a non-zero entry

¹Specifically, let \mathbf{R}_1 be a global randomizer and let \mathbf{R}_2 be a local randomizer. Then, $\mathbf{R} = \mathbf{R}_1\mathbf{R}_2$.

with either -1 or $+1$ and \mathbf{R} costs $O(N)$ adds. In sum, the computation cost is $O(N + M \log M)$.

Furthermore, in terms of storage cost, \mathbf{D} is equivalent to $\text{circ}(\mathbf{d}_0)$ and saving \mathbf{d}_0 costs $O(M)$. Φ is not necessary to be saved since Φ , in fact, is finished by:

$$(\Phi\mathbf{R}\mathbf{x})_k = \sum_{i=0}^{\frac{N}{M}-1} (\mathbf{R}\mathbf{x})_{k+iM}. \quad (3)$$

\mathbf{R} only has N non-zero entries and costs $O(N)$. Thus, the total storage cost is $O(N)$.

Table 1 depicts the comparison between our scheme and representative fast embedding methods. Specifically, \mathbf{A} 's in [3] and [7] are designed as a Gaussian random matrix and circulant matrix, respectively. [6] reshapes \mathbf{x} into two-dimensional data, which are projected by two separable Gaussian random matrices with smaller size.

Our approach exhibits the best desired requirement in terms of computation and storage costs. In addition, when one only focuses on the number of multiplications (adds can be handled more efficiently than multiplications) [12], our scheme only requires $O(M \log M)$ computation cost.

Table 1. Comparison of computation and storage costs for data-independent binary embedding methods.

Methods	Computation	Storage
Full projection [3]	$O(MN)$	$O(MN)$
Bilinear proj. [6]	$O(N^{1.5})$	$O(N)$
Circulant proj. [7]	$O(N \log N)$	$O(N)$
Our scheme	$O(N + M \log M)$	$O(N)$

3.2. Angle-Preserving Property Based on Sparsity

Like [3][7][13], we analyze the property of similarity (angle)-preserving for the proposed scheme in this section. Angle-preserving is useful because angle includes the information about similarity between data, which is an important physical property in many applications, including image retrieval and nearest neighbor search.

Suppose $\mathcal{H}_M(\mathbf{x}_1, \mathbf{x}_2)$ is the normalized Hamming distance between $\mathbf{x}_1, \mathbf{x}_2$:

$$\mathcal{H}_M(\mathbf{x}_1, \mathbf{x}_2) = \frac{1}{2M} \sum_{i=0}^M |\text{sign}(\mathbf{a}^i \mathbf{x}_1) - \text{sign}(\mathbf{a}^i \mathbf{x}_2)|, \quad (4)$$

where \mathbf{a}^j is the j^{th} row of \mathbf{A} . It is expected that $\mathcal{H}_M(\mathbf{x}_1, \mathbf{x}_2)$ is related to the angle θ between \mathbf{x}_1 and \mathbf{x}_2 . The ideal case of angle-preserving property satisfies $E\{\mathcal{H}_M(\mathbf{x}_1, \mathbf{x}_2)\} = c\theta$, where c is a constant, and $\text{Var}\{\mathcal{H}_M(\mathbf{x}_1, \mathbf{x}_2)\} = 0$.

If \mathbf{A} is drawn from i.i.d distribution, which collides with the proposed method, M. S. Charikar [3] has shown $E\{\mathcal{H}_M(\mathbf{x}_1, \mathbf{x}_2)\} = \frac{\theta}{\pi}$ and $\text{Var}\{\mathcal{H}_M(\mathbf{x}_1, \mathbf{x}_2)\} = \frac{\theta(\pi-\theta)}{M\pi^2}$.

Chang *et al.* [7] only show by experiments if \mathbf{A} is a circulant matrix, whose first row is a Gaussian random vector, the *sample* mean and *sample* variance of $\mathcal{H}_M(\mathbf{x}_1, \mathbf{x}_2)$ corresponding to \mathbf{A} approximates the results of M. S. Charikar [3].

Our proof of angle-preserving property includes two steps: 1) Let $\hat{\mathbf{x}} = \Phi \mathbf{R} \mathbf{x}$. We prove \mathbf{D} can preserve the angle $\hat{\theta}$ between $\hat{\mathbf{x}}_1$ and $\hat{\mathbf{x}}_2$. 2) Then, we show $\hat{\theta} \sim \theta$ holds, which implies our scheme preserves θ .

For the first step, [7] has validated if \mathbf{d}^0 is a Gaussian random vector, then \mathbf{D} preserves $\hat{\theta}$ between $\hat{\mathbf{x}}_1$ and $\hat{\mathbf{x}}_2$ after embedding. For the second step, Chang and Wu [14] show that if a matrix satisfies δ_K -RIP, it also preserves angle with the distortion being proportional to δ_K after embedding.

Theorem 1. (δ_K -RIP [15]) Let $\mathbf{A} \in \mathbb{R}^{M \times N}$ be a random matrix drawn according to any distribution that satisfies the concentration inequality. Then, for any K -sparse signal \mathbf{x} and any $0 < \delta_K < 1$, we have

$$(1 - \delta_K) \|\mathbf{x}\|_2^2 \leq \|\mathbf{A} \mathbf{x}\|_2^2 \leq (1 + \delta_K) \|\mathbf{x}\|_2^2, \quad (5)$$

with the probability

$$\geq 1 - 2 \left(\frac{12}{\delta_K} \right)^K e^{-\left(\frac{\delta_K^2}{16} - \frac{\delta_K^3}{48} \right) M}.$$

We call a matrix satisfying δ_K -RIP when Eq. (5) holds. In other words, if $\Phi \mathbf{R}$ satisfies δ_K -RIP, $\Phi \mathbf{R}$ preserves the angle. To date, finding a deterministic matrix satisfying RIP within polynomial time, however, is still an open problem [16]. Unfortunately, the proposed projection matrix is deterministic to violate Theorem 1.

To overcome this problem, we derive another theoretical bound about δ_K along with the lower bound of the probability. We start from the following Lemma.

Lemma 1. Let $\mathbf{x} \in \mathbb{R}^N$ be K -sparse, let $(\hat{\mathbf{x}})_k = \sum_{i \in S_k} (\mathbf{R} \mathbf{x})_i$ with $k = 0, 1, \dots, M-1$, let $S_k = \{k + jM | (\mathbf{R} \mathbf{x})_{k+jM} \neq 0 \text{ for } j = 0, \dots, \frac{N}{M} - 1\}$, and let $\kappa_k = |S_k|$. Then,

$$|\{k | \kappa_k \geq 2 \text{ for } k = 0, \dots, M-1\}| < f,$$

hold for $f = 1, \dots, \frac{K}{2}$ with the probability being larger than

$$\geq 1 - \binom{M}{f} \left(\frac{\frac{N}{M}}{2} \right)^f \binom{N-2f}{K-2f} / \binom{N}{K}.$$

Moreover, by Stirling's formula, the bound is relaxed into

$$\geq 1 - \frac{1}{\sqrt{2\pi f}} \left(\frac{eK^2}{2Mf} \right)^f.$$

Proof. To simplify the notation, let E_f be the event with $|\{k | \kappa_k \geq 2 \text{ for } k = 0, \dots, M-1\}| < f$. The event is related to the positions of non-zero entries of $\mathbf{R} \mathbf{x}$ but is unrelated to their values. Since \mathbf{R} permutes \mathbf{x} randomly, the positions

of non-zero entries of $\mathbf{R} \mathbf{x}$ are uniformly distributed. Thus, $P\{E_f\}$ is considered as a combination problem. Let $\binom{N}{K}$ be all combinations of N positions taking K positions being non-zeros at a time. Then, $P\{E_f\}$ is equal to divide the number of combinations belonging to E_f by $\binom{N}{K}$.

Instead of calculating $P\{E_f\}$ directly, we focus on $P\{E_f^c\}$, which is the complement of E_f . Specifically, E_f^c is the event with $|\{k | \kappa_k \geq 2 \text{ for } k = 0, \dots, M-1\}| \geq f$. Then,

$$P\{E_f^c\} \leq \binom{M}{f} \left(\frac{\frac{N}{M}}{2} \right)^f \binom{N-2f}{K-2f} / \binom{N}{K}.$$

$\binom{M}{f} \left(\frac{\frac{N}{M}}{2} \right)^f$ means choosing f sets from S_0, S_1, \dots, S_{M-1} such that the chosen sets satisfy $\kappa_k = 2$. Thus, $2f$ non-zero entries of \mathbf{x} are arranged. Then, $\binom{N-2f}{K-2f}$ means the remaining $(K-2f)$ non-zero entries of \mathbf{x} distribute randomly among the remaining $N-2f$ positions.

Consequently, since $P\{E_f\} + P\{E_f^c\} = 1$, we have $P\{E_f\} = 1 - P\{E_f^c\} \geq 1 - \binom{M}{f} \left(\frac{\frac{N}{M}}{2} \right)^f \binom{N-2f}{K-2f} / \binom{N}{K}$.

Further, the term $\binom{M}{f} \left(\frac{\frac{N}{M}}{2} \right)^f \binom{N-2f}{K-2f} / \binom{N}{K}$ is approximated by:

$$\begin{aligned} & \binom{M}{f} \left(\frac{\frac{N}{M}}{2} \right)^f \binom{N-2f}{K-2f} / \binom{N}{K} \\ & \leq \frac{N^{2f}}{f!(2M)^f} \frac{(N-2f)!K!}{(K-2f)!N!} \\ & \leq \frac{K^{2f}}{f!(2M)^f} \frac{(N-2f)!N^{2f}}{N!} \\ & \sim \frac{1}{\sqrt{2\pi f}} \left(\frac{eK^2}{2Mf} \right)^f. \end{aligned}$$

The last deviation is due to $f! \sim \sqrt{2\pi f} \left(\frac{f}{e} \right)^f$ by Stirling's formula, where the approximation is more accurate when N is large enough. Thus, $P\{E_f\} \geq 1 - \frac{1}{\sqrt{2\pi f}} \left(\frac{eK^2}{2Mf} \right)^f$. We complete this proof. \square

It should be noted that, if $\kappa_k = 1$, it implies $(\hat{\mathbf{x}})_k$ is equal to one of non-zero entries of $\mathbf{R} \mathbf{x}$. If $\kappa_k = 1$ or 0 for $0 \leq k \leq M-1$, it means no distance distortion and $\|\hat{\mathbf{x}}\|_2 = \|\mathbf{R} \mathbf{x}\|_2 = \|\mathbf{x}\|_2$. Thus, based on Lemma 1, we can derive in Theorem 2 the probability with $\delta_K = 0$.

Theorem 2. Let $\Phi \mathbf{R} \in \mathbb{R}^{M \times N}$. Then, for any K -sparse \mathbf{x} , we have $\delta_K = 0$ such that

$$\|\Phi \mathbf{R} \mathbf{x}\|_2^2 = \|\mathbf{x}\|_2^2, \quad (6)$$

with the probability

$$\geq 1 - M \binom{\frac{N}{M}}{2} \binom{N-2}{K-2} / \binom{N}{K}.$$

Moreover, by Stirling's formula, the bound is relaxed into

$$\geq 1 - \frac{1}{\sqrt{2\pi}} \left(\frac{eK^2}{2M} \right).$$

Proof. Following the same notations in the proof of Lemma 1, let E_1 be the event with $|\{k | \kappa_k \geq 2 \text{ for } k = 0, \dots, M-1\}| < 1$. In other words, if E_1 occurs, it means $\kappa_k = 1$ or 0 for $0 \leq k \leq M-1$. Thus, $P\{\|\Phi \mathbf{R} \mathbf{x}\|_2^2 = \|\mathbf{x}\|_2^2\} = P\{E_1\}$. $P\{E_1\} \geq 1 - M \binom{\frac{N}{M}}{2} \binom{N-2}{K-2} / \binom{N}{K}$ is calculated by setting $f = 1$ in Lemma 1. We complete this proof. \square

Theorem 2 indicates that, if $K \leq O(\sqrt{M})$, the probability of $\delta_K = 0$ is high enough. We will validate Theorem 2 by experiments later.

We further extend Theorem 2 to consider different values of δ_K . Nevertheless, if $\kappa_k > 1$, $(\Phi \mathbf{R} \mathbf{x})_k$ is the sum of at least two non-zero entries of \mathbf{x} . In this case, different signals (\mathbf{x} 's) will led to different distance distortions. To simplify the problem, we assume $\mathbf{x} \in \{0, 1\}^N$. Under the circumstance, theoretical bound for δ_K is derived in Theorem 3.

Theorem 3. Let $\Phi \mathbf{R} \in \mathbb{R}^{M \times N}$ and $g = \left(\frac{N}{M}\right)$. Then, for any K -sparse $\mathbf{x} \in \{0, 1\}^N$ and any $\delta_K \in \{0, \frac{2g}{K}, \frac{4g}{K}, \dots, g - \frac{2g}{K}\}$, we have

$$(1 - \delta_K) \|\mathbf{x}\|_2^2 \leq \|\Phi \mathbf{R} \mathbf{x}\|_2^2 \leq (1 + \delta_K) \|\mathbf{x}\|_2^2, \quad (7)$$

with the probability

$$\geq 1 - \binom{M}{f} \binom{\frac{N}{M}}{2}^f \binom{N-2f}{K-2f} / \binom{N}{K},$$

where $f = \frac{\delta_K K}{2g} + 1$. Moreover, by Stirling's formula, the bound is relaxed into

$$\geq 1 - \frac{1}{\sqrt{2\pi f}} \left(\frac{eK^2}{2Mf} \right)^f.$$

Proof. We use the same notation and definition in Lemma 1. If E_f occurs, without loss of generality, let $\kappa_i \geq 2$ for $i = 0, \dots, f-2$ and $\kappa_j = 1$ for $j = f-1, \dots, M-1$. Then,

$$\begin{aligned} \|\Phi \mathbf{R} \mathbf{x}\|_2^2 &\leq (1 + \delta_K) \|\mathbf{x}\|_2^2 \\ \Rightarrow \delta_K &= \max_{\mathbf{x}} \frac{\|\Phi \mathbf{R} \mathbf{x}\|_2^2 - \|\mathbf{x}\|_2^2}{\|\mathbf{x}\|_2^2} \\ \Rightarrow \delta_K &= \max_{\mathbf{x}} \frac{\sum_{k=0}^{f-2} \sum_{i \in S_k, j \in S_k/i} 2(\mathbf{R} \mathbf{x})_i (\mathbf{R} \mathbf{x})_j}{\|\mathbf{x}\|_2^2} \quad (8) \\ \Rightarrow \delta_K &= \frac{2g(f-1)}{K} \end{aligned}$$

The derivation in last line of Eq. (8) comes from the fact that the non-zero entries of \mathbf{x} are 1. Thus, $2(\mathbf{R} \mathbf{x})_i (\mathbf{R} \mathbf{x})_j$ has maximal value 2. Further, the cardinality of $\{(i, j) | i \in S_k, j \in S_k/i\}$ is $\binom{\kappa_k}{2}$. In the worst case, $\kappa_k = \frac{N}{M}$. Thus,

$$\sum_{i \in S_k, j \in S_k/i} 2(\mathbf{R} \mathbf{x})_i (\mathbf{R} \mathbf{x})_j = \sum_{i \in S_k, j \in S_k/i} 2 = 2 \binom{\frac{N}{M}}{2} = 2g.$$

Consequently, $\frac{2g(f-1)}{K} = \delta_K$ or $f = \frac{\delta_K K}{2g} + 1$. If $f = 1$, it implies that $0 = \delta_K$ with the probability $P\{E_1\}$, that is a special case like Theorem 2. Since $f \in \{1, 2, \dots, \frac{K}{2}\}$, we have $\delta_K \in \{0, \frac{2g}{K}, \frac{4g}{K}, \dots, g - \frac{2g}{K}\}$ along with the corresponding probability $P\{E_f\} = P\{E_{\frac{\delta_K K}{2g} + 1}\}$. We complete this proof. \square

We want to briefly discuss why we assume $\mathbf{x} \in \{0, 1\}^N$ instead of other signal types such as Gaussian random signal. The larger $\sum_{k=0}^{f-1} \sum_{i \in S_k, j \in S_k/i} 2(\mathbf{R} \mathbf{x})_i (\mathbf{R} \mathbf{x})_j$ is, the large δ_K is. Thus, assuming \mathbf{x} has constant energy such that $\|\mathbf{x}\|_2 = c$, the largest δ_K is equivalent to solving the following optimization problem:

$$\begin{aligned} \max_{\mathbf{x}} \sum_{k=0}^{f-1} \sum_{i \in S_k, j \in S_k/i} 2(\mathbf{R} \mathbf{x})_i (\mathbf{R} \mathbf{x})_j \quad (9) \\ \text{subject to } \|\mathbf{x}\|_2 = c. \end{aligned}$$

By solving the optimization problem by Lagrange multiplier, the optimal value is achieved with the constraint that $(\mathbf{R} \mathbf{x})_i = (\mathbf{R} \mathbf{x})_j$ with $i, j \in S_k$ for $k = 0, \dots, f-1$. If \mathbf{R} is a deterministic matrix, it is easy to obtain optimal solution \mathbf{x} . However, \mathbf{R} is a randomizer resulting in random locations and random sign of \mathbf{x} . By assuming $\mathbf{x} \in \{0, 1\}^N$, $(\mathbf{R} \mathbf{x})_i = (\mathbf{R} \mathbf{x})_j$ holds with high probability. We emphasize that rigorous proof is still absent and should be discussed in the future work.

To check whether Φ is good enough to satisfy δ_K -RIP from empirical and theoretical results, we compare it with Gaussian random matrix, which is admitted to be a good choice for satisfying δ_K -RIP. Let \mathbf{A} be designed as either a Gaussian random matrix drawn from $\mathcal{N}(0, \frac{1}{M})$ or the proposed projection matrix. A Monte Carlo method is used to estimate RIP. By generating a set of K -sparse signals (*i.e.*, \mathbf{x} 's), where non-zero entries are 1's, $E\{\delta_K\}$ can be estimated. Table 2 shows the empirical results, where each one is obtained from the mean of 100,000 trials. The proposed matrix benefits from the sparsity property and outperforms Gaussian random matrix. Basically, the simulation results actually meet the theoretical prediction. Moreover, Table 3 shows the case that non-zero entries of \mathbf{x} are drawn from $\mathcal{N}(0, 1)$. We can see that δ_K 's are smaller than those in Table 2.

In addition, the lower bound of probability of satisfying δ_K -RIP in Theorem 3 is tighter than that in Theorem 1, as

shown in Fig. 1, where solid curves denote the empirical results generated by Monte Carlo method and dash curves denote the corresponding theoretical lower bounds based on Theorem 1 and Theorem 3. Fig. 1 reveals that the lower bound in Theorem 1 is not trivial only when \mathbf{x} is very sparse. Otherwise, it is always zero. Fig. 2 shows the histogram of δ_K under different settings of N , M , and K . The horizontal axis in Fig. 2(b) is discrete because of $\mathbf{x} \in \{0, 1\}^N$. In sum, the proposed projection matrix has a higher probability to satisfy δ_K -RIP with small δ_K .

Consequently, since our designed $\Phi\mathbf{R}$ can satisfy δ_K -RIP, it also preserves similarity between two data, as proved in [14]. Combined with the fact that $\mathbf{D} = \text{circ}(\mathbf{d}^0)$, with \mathbf{d}^0 being a Gaussian random vector, also preserves the angle between two data [7], our proposed $\mathbf{D}\Phi\mathbf{R}$ still retains angle-preserving property.

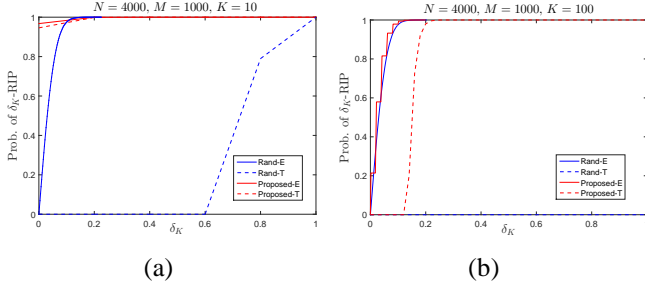


Fig. 1. Probability of satisfying δ_K -RIP versus δ_K when \mathbf{A} is either the proposed matrix or a Gaussian random matrix. Proposed-E and Rand-E denote empirical results while Proposed-T and Rand-T denote the lower bounds of probability in Theorem 3 and Theorem 1, respectively. (a) $N = 4000$, $M = 1000$, $K = 10$. (b) $N = 4000$, $M = 1000$, $K = 100$.

Table 2. Estimation of δ_K for \mathbf{A} being either a Gaussian random matrix or the proposed projection matrix under $N = 4000$ and different M and K . The result is presented by a/b , where a and b denote $E\{\delta_K\}$'s obtained by Gaussian random matrix and the proposed matrix, respectively. Bold represents the better results.

M \ K	25	50	100	200	400
1000	.035/. 015	.036/. 025	.036/. 028	.037/. 030	.037/. 031
500	.048/. 031	.049/. 040	.050/. 044	.050/. 045	.051/. 046
250	.070/. 055	.070/. 065	.069/. 066	.071/. 067	.073/. 069
125	.101/. 093	.100/. 095	.105/. 097	.101/. 098	.101/. 100

4. EXPERIMENTAL RESULTS

Simulations were conducted in Matlab environment with an Intel CPU Q6600 and 16 GB RAM under MS Win7 (64 bits). Since we focus on the comparison of computation and storage costs, we only compare the proposed algorithm with some

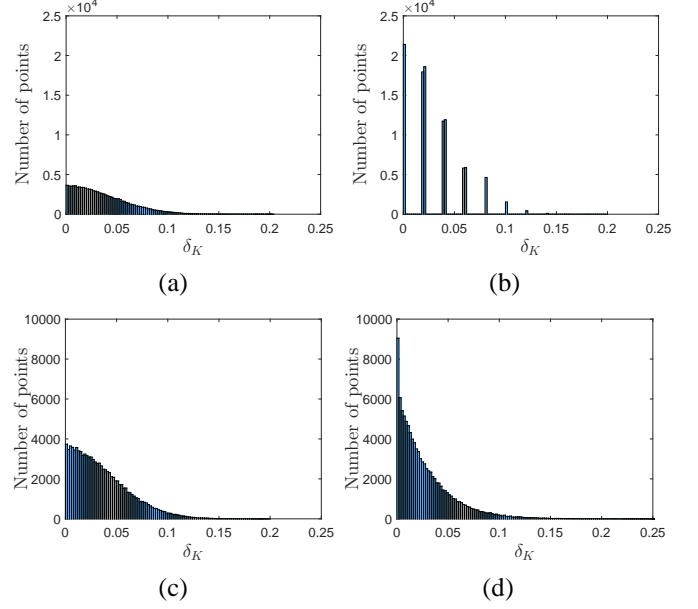


Fig. 2. Histogram for density estimation of δ_K with $N = 4000$, $M = 1000$, and $K = 100$. (a)(b) $\mathbf{x} \in \{0, 1\}^N$. (c)(d) \mathbf{x} is drawn from $\mathcal{N}(0, 1)$. (a)(c) Gaussian random matrix. (b)(d) The proposed matrix.

Table 3. Estimation of δ_K for \mathbf{A} being either a Gaussian random matrix or the proposed projection matrix. Except that the non-zero entries of \mathbf{x} are drawn from $\mathcal{N}(0, 1)$, other settings follow Table 2.

M \ K	63	125	250	500	1000
1000	.033/. 011	.035/. 018	.037/. 026	.036/. 028	.036/. 030
500	.048/. 025	.050/. 035	.050/. 039	.050/. 042	.051/. 044
250	.070/. 047	.068/. 061	.069/. 063	.072/. 065	.072/. 068
125	.101/. 080	.102/. 091	.101/. 094	.101/. 096	.101/. 097

selected data-independent binary embedding algorithms, including

- Locality Sensitive Hashing (LSH) [3]: \mathbf{A} is a Gaussian random matrix. This method is considered as a baseline in terms of performance and computation cost.
- CBE-rand [7]: \mathbf{A} is designed as a circulant matrix, where the seed vector is a Gaussian random vector. This method focuses on fast embedding by FFT.
- BP-rand [6]: Use two matrices to separably project data. We follow the data-independent setting in [6], where two matrices are designed as Gaussian random matrices without learning.

Except the proposed method and BP-rand, all other codes were downloaded from <http://www.unc.edu/~yunchao/>. According to the following evaluations, our method is concluded

to be very efficient to compute binary codes with low memory requirements and exhibit performance of image classification and retrieval being comparable to state-of-the-art data-independent projection techniques.

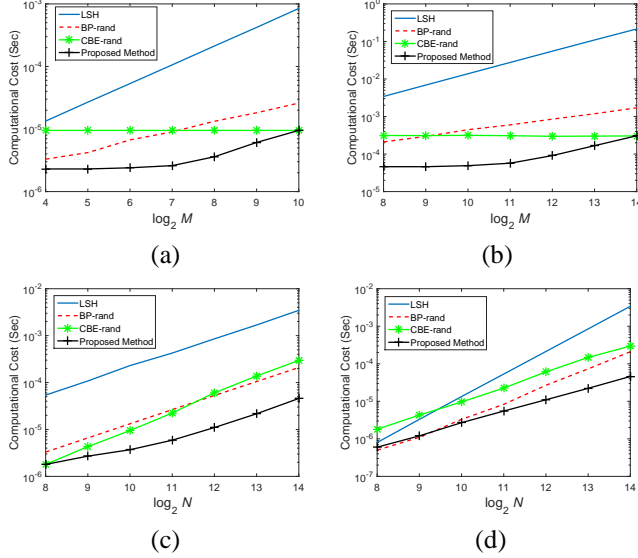


Fig. 3. Comparisons between different approaches in terms of computation cost. (a) Fixing $N = 2^{10}$, M versus computation time. (b) Fixing $N = 2^{14}$, M versus computation time. (c) Fixing $M = 2^8$, N versus computation time. (d) Fixing the compression ratio $\frac{N}{M} = 2^6$, N versus computation time.

4.1. Computation and Memory Costs

Since computation cost are invariant to signal types, synthesis data were used here. Storage cost is equal to the memory requirement for saving projection \mathbf{A} . Figs. 3(a) and (b) show the computation time versus different M 's under $N = 2^{10}$ and $N = 2^{14}$, respectively. Fig. 3(c) shows the results obtained from different N 's under $M = 2^8$. One can clearly find that the proposed method outperforms the other methods (note the logarithmic scale of the vertical axis). We can validate the experimental results along with theoretical results in Table 1. When $M = N$, the computation cost of our method is equal to that of CBE-rand. When $M < N$, we have two observations from Figs. 3(a)~(c): (i) our method is dominated by $O(N)$ when $M \log M < N$ and (ii) $O(M \log M)$ dominates the computation cost when $M \log M \geq N$.

In addition, fixing the compression ratio $\frac{N}{M}$, we have $M \log M > N$ for sufficiently large N . It implies that for high-dimensional signals with a fixed compression ratio, the proposed method speeds up projection remarkably. Fig. 3(d) further shows the computation cost of our method increases slower than other methods with constant compression ratio. It should be noted that BP-rand outperforms CBE-rand and our method when $N \leq 2^9$ because (i) $N^{1.5}$ approximates

$N \log N$ when N is small and (ii) CBE-rand and our method incur larger Big-O constants due to the use of FFT.

On the other hand, Table 4 shows the comparison of memory cost for saving projection. We follow the parameter setting in [6] with $M = N$. It is observed that our method is nearly comparable to CBE-rand.

However, our method actually requires less memory and outperforms CBE-rand under practical scenario with $M < N$, as depicted in Table 5. This is because the cost of Φ in our method only depends on M but that in CBE-rand depends on N .

Table 4. Memory (MegaBytes) needed to store the projection matrix, assuming each element is float-point (32 bits). Note the results of BP-rand is directly copied from Table 2 of [6].

N	LSH	BP-rand	CBE-rand	Ours
1.28×10^3	6.25	0.06	0.0049	0.005
1.28×10^4	625	0.10	0.049	0.0504
2.56×10^4	2500	0.22	0.0977	0.1007
6.4×10^4	15625	1.02	0.2441	0.2518
1.28×10^5	62500	3.88	0.4883	0.5035

Table 5. Memory (MegaBytes) required for our method and CBE-rand under fixed $N = 1.28 \times 10^5$ and various M .

M	1.6×10^4	3.4×10^4	6.4×10^4	1.28×10^5
CBE-rand	0.4883	0.4883	0.4883	0.4883
Ours	0.0763	0.1373	0.2594	0.5035

4.2. Image Applications

We verify whether binary codes yielded after our embedding scheme, despite its low computation and storage cost, still contain discriminative power in image classification and retrieval.

4.2.1. Image Classification

Two datasets were considered in image classification:

- CIFAR [17]: It consists of 64,800 images that have been manually grouped into 11 ground-truth classes (airplane, automobile, bird, boat, cat, deer, dog, frog, horse, ship and truck). All images were represented as GIST descriptor [18] with $N = 2048$.
- MNIST [19]: It includes 60,000 images with handwriting digits from 0 – 9. All images were represented as GIST descriptor [18] with $N = 512$.

After embedding, binary codes were fed into LIBSVM [20] to train classifier by supervised learning (8-fold cross-validation). Ground truth is based on pre-defined labels provided by the datasets. Fig. 4 shows the accuracy versus different M bits, where accuracy is the probability that classifier

has labeled an testing image into the ground truth. In both CIFAR and MNIST datasets, the proposed method is comparable to LSH and CBE-rand, but the performance of BP-rand degrades due to projection within a bilinear structure. In addition, though GIST feature is not sparse, our method still exhibits good performance because the features are still approximately sparse, where only few entries are significant.

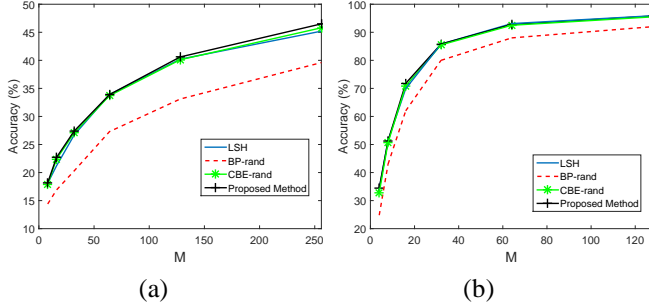


Fig. 4. Accuracy vs. M . Classifier is learned by LIBSVM. (a) CIFAR dataset (b) MNIST dataset.

4.2.2. Image Retrieval

For purpose of image retrieval, we used the same datasets and setting in image classification. All images still were represented by GIST features. In this experiment, “retrieval” was performed by randomly selecting 1,000 query images from dataset and returning images according to hamming distance sorting in an ascending order. Performance is measured by mean Average Precision (mAP) [8].

Fig. 5 shows mAP with top 50 returned images. Whatever M is, the proposed approach has the comparable performance with LSH and CBE-rand. In other words, the proposed method preserves angle (similarity) well even an extra downsampling matrix is introduced to achieve faster binary embedding.

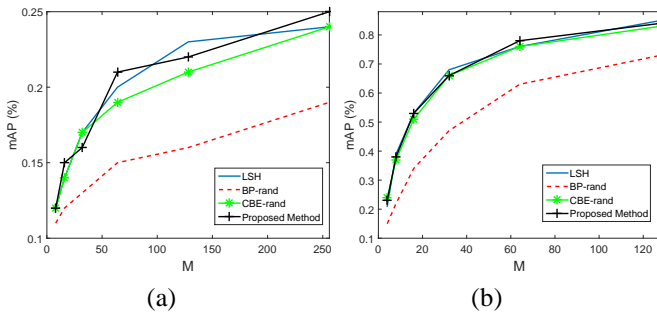


Fig. 5. mAP vs. M with top 50 returned images. (a) CIFAR dataset. (b) MNIST dataset.

5. CONCLUSIONS AND FUTURE WORKS

In this paper, we have proposed a data-independent binary embedding technique with $O(N + M \log M)$ in computation cost and $O(N)$ in storage cost to outperform state-of-the-art approaches. We also theoretically prove that if data have sparsity, similarity (angle) between data is preserved well. The full potential of our method is applied for ultra-high dimensional data [7], for which no other methods are applicable.

For future work, the goal is to extend our method to data-dependent paradigm. That is, given R , $\Phi R x$ is considered to be new training data instead of x . All we need to do is to learn a circulant matrix D . Thus, the learning process applies to low-dimensional data ($\Phi R x$), resulting in low computation and memory costs. After that, our goal is to simultaneously learn D and R .

6. ACKNOWLEDGMENT

This work was supported by Ministry of Science and Technology, Taiwan, ROC, under grants MOST 104-2221-E-001-019-MY3 and 104-2221-E-001-030-MY3.

7. REFERENCES

- [1] L.-K. Huang, Q. Yang, and W.-S. Zheng, “Online hashing,” in *Proceedings of the international joint conference on Artificial Intelligence*, pp. 1422–1428, 2013.
- [2] C Leng, J. Wu, J. Cheng, X. Bai, and H. Lu, “Online sketching hashing,” *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2503–2511, 2015.
- [3] M. S. Charikar, “Similarity estimation techniques from rounding algorithms,” *ACM Symposium on Theory of Computing*, pp. 380–388, 2002.
- [4] M. Raginsky and S. Lazebnik, “Localitysensitive binary codes from shift-invariant kernels,” *Neural Information Processing Systems*, 2009.
- [5] L.-W. Kang and C.-S. Lu, “Compressive sensing-based image hashing,” *IEEE Conference on Image Processing*, pp. 1285–1288, 2009.
- [6] Y. Gong, K. Sanjiv, H. A. Rowley, and S. Lazebnik, “Learning binary codes for highdimensional data using bilinear projections,” *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 484–491, 2013.
- [7] F. Yu, S. Kumar, Y. Gong, and S.-F. Chang, “Circulant binary embedding,” in *International Conference on Machine Learning*, 2014.
- [8] Y. Gong, S. Lazebnik, A. Gordo, and F. Perronnin, “Iterative quantization: A procrustean approach to learning binary codes for large-scale image retrieval,” *IEEE*

Transactions on Pattern Analysis and Machine Intelligence, vol. 35, pp. 2916–2929, 2013.

- [9] Y. Xia, K. He, P. Kohli, and J. Sun, “Compressive signal processing with circulant sensing matrices,” *IEEE international conference on Acoustic, Speech and Signal Processing*, pp. 1015–1019, 2015.
- [10] Y. Pan H. Lai, and, Y. Liu, and S. Yan, “Simultaneous feature learning and hash coding with deep neural networks,” *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3270–3278, 2015.
- [11] P. Boufounos and R. G. Baraniuk, “1-bit compressive sensing,” *Conf. on Info. Sciences and Systems*, pp. 16–21, 2008.
- [12] H. Hassanieh, P. Indyk, D Katabi, and Eric Price, “Faster gps via the sparse fourier transform,” in *ACM MOBICOM*, 2012.
- [13] S. Kim and S. Choi, “Bilinear random projections for locality-sensitive binary codes,” *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1338 – 1346, 2015.
- [14] L.-H. Chang and J.-Y. Wu, “Achievable angles between two compressed sparse vectors under norm/distance constraints imposed by the restricted isometry property: A plane geometry approach,” *IEEE Transactions on Information Theory*, vol. 59, pp. 2059–2081, 2013.
- [15] R. Baraniuk, M. Davenport, R. Devore, and M. Wakin, “A simple proof of the restricted isometry property for random matrices,” *Constructive Approximation*, vol. 28, pp. 253–263, 2008.
- [16] S. Foucart and H. Rauhut, “A mathematical introduction to compressive sensing,” in *Applied and Numerical Harmonic Analysis*, 2013.
- [17] A. Krizhevsky, “Learning multiple layers of features from tiny images,” *Tech Report. University of Toronto*, 2009.
- [18] A. Torralba A. Oliva, and, “Modeling the shape of the scene: A holistic representation of the spatial envelope,” *International Journal of Computer Vision*, vol. 42, pp. 145–175, 2001.
- [19] Y. LeCun and C. Cortes, “The mnist database of handwritten digits,” 1998.
- [20] C. C. Chang and C.-J. Lin, “Libsvm: A library for support vector machines,” *ACM Transactions on Intelligent Systems and Technology*, vol. 2, pp. 1–27, 2011.